

Amendments to the Specification:

Please amend the specification as follows:

Third Paragraph of page 26, lines 19-24 and first paragraph on page 27, lines 1-10:

A1
As to the organization of the dynamic flat file system 300 of Figure 3A, in one embodiment, a structure of the dynamic flat file system is stored as a special file (i.e., file system metadata). In one embodiment, the dynamic flat file system metadata is stored in a first logical cylinder of a disk volume. In another embodiment, a duplicate of the dynamic flat file system metadata is stored in the 0X100 binary (i.e., 256 in decimal scale) logical cylinder. Substantially the entire of both of the logical cylinders is reserved for the storage of the dynamic flat file system metadata. The structure of the dynamic flat file system stored as a special file contains the contents of substantially the entire portion of the dynamic flat file system that is in use. As such, the structure does not represent existing free space. However, the volume of free storage space can be calculated and its hexadecimal value can be stored on a logical disk volume. The benefits of not representing free storage space in the dynamic flat file system metadata of the present invention are twofold. One is that the size of the dynamic flat file system metadata is substantially minimized. Second, it is not necessary to recreate the entire structure of the dynamic flat file system in order to increase the amount of its free storage space, as this task can simply be achieved by changing the calculation of the volume of the free storage.

Third paragraph on page 35, lines 22-24, and first paragraph on page 36, lines 1-13:

A2
Figure 7 is a high level diagram of an independent storage system 700 including a plurality of consumer nodes having heterogeneous ~~files~~ file systems in communication with a plurality of I/O nodes via a transport, in accordance with one embodiment of the present invention. As shown, the independent storage system 700 includes consumer nodes (i.e.,

A2
requesters) 602a, 602b, and 602c, each of which is coupled to a transport 606 via an associated transport connection 602a', 602b', and 602c'. Typically, the consumer nodes 602a through 602c can be personal computers, servers, personal digital assistants "PDA," or any other computing device having a processor, memory, and transport hardware for communicating over a transport. For example, in one implementation, personal computers are coupled to the transport via a transport hardware, which a user or process uses to store and retrieve needed data. As shown, each of the consumer nodes 602a, 602b, and 602c has an associated consumer computer logic 603a, 603b, and 603c. In one exemplary embodiment, each of the consumer computer logics 603a, 603b, and 603c is configured to include an application layer, a file system layer, and an independent storage driver. Additional details with respect to the components of the consumer computer logic are described below in connection with Figure 8.

Third paragraph on page 14, lines 9-22:

A3
Figure 1 is a flow chart 100 illustrating an exemplary method of creating a dynamic flat file system, in accordance with one embodiment of the present invention. The method begins at an operation 102 wherein a set of file systems is provided. Exemplary file systems are those associated with operating systems such as, UnixTM, Sun Microsystems Inc. SolarisTM, Microsoft Corp. WindowsTM, Apple Computer Inc. Mac OSTM, etc. These file systems include, for example, FAT16, FAT32, NTFS, HPFS, UFS, EFS, Berkeley FS, AT&T Unix FS, LFS, AFS, Unicos FS, etc. The method then moves to an operation 104 where metadata of each of the provided file systems is examined. This examination begins by operation 104a wherein a first metadata type is identified. Then, the method moves to operation 104b where a determination is made as to what the particular metadata being examined means in the context of a specific file system. That is, a determination is made as to what information the metadata intends to convey and why the metadata conveys such an information. Next, in operation 104c, the method maps meaning to behavioral model.
